Controlador de Interrupciones
(Versión programable)
Manual de Usuario

Índice de contenido

1. Características

- El controlador implementa el protocolo “Daisy Chain” de Zilog.
- Compatible con el microprocesador T80.
- Vector de interrupciones programable por el usuario.
- Posibilidad de leer el estado del controlador por software.
- Permite el borrado de solicitudes de interrupción pendientes.
- Capacidad de recordar una solicitud hecha por un periférico mientras es atendida una solicitud previa.
- Salidas para dar reconocimiento de interrupción a periféricos diseñados para modo 1.

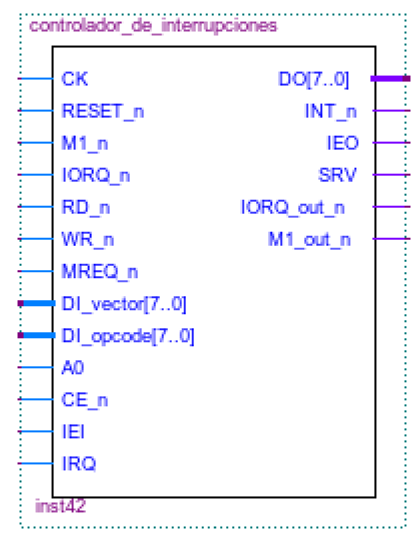


Fig. 1 Esquema del controlador con sus pines de conexión.

2. Descripción general

El controlador de interrupciones se utiliza conectado a un periférico a través de su entrada IRQ, de un bit de ancho, que permite comunicar la solicitud de una interrupción. Cuando se arma un sistema con varios controladores (conectados a sus correspondientes periféricos) en Daisy Chain, el manejo de las señales IEI e IEO por parte de los mismos permite el arbitraje de interrupciones, según el protocolo definido por Zilog.

El usuario puede realizar por software las siguientes operaciones:

- Escritura/lectura del vector de interrupciones.
- Lectura del estado del controlador codificado en un byte.
- Borrado de peticiones pendientes (mediante escritura a una dirección de E/S).

Para operaciones de lectura/escritura el controlador posee puertos de entrada/salida. Una

entrada de habilitación (CE_n) permite realizar externamente la decodificación y el bit menos significativo del bus de direcciones (A0) es utilizado para seleccionar el tipo de operación. Existe una versión del controlador en la cual el vector de interrupciones es configurable por el usuario durante el armado del sistema.

3. Descripción funcional

Cuando el periférico genera un flanco creciente en la entrada IRQ, es registrada una solicitud en un flip flop interno (IFF) y se pone IEO = '0'. Si lo permite la entrada IEI (= '1'), se baja la salida $\overline{\text{INT}}$ y se espera al reconocimiento. Cuando el microprocesador realiza un ciclo de reconocimiento e IEI = '1' el controlador coloca el vector de interrupciones en el bus de entrada al microprocesador. Si dentro del ciclo de reconocimiento la entrada IEI baja a '0', es retirado el vector del bus. De este modo es atendido el problema de la propagación IEI – IEO entre controladores.

Mientras se está ejecutando la subrutina de atención de un periférico, éste puede solicitar una nueva interrupción. Esta nueva solicitud es registrada y atendida al finalizar la subrutina, si los periféricos de mayor prioridad lo permiten. El usuario puede borrar una nueva solicitud mientras la anterior está siendo atendida.

La salida SRV permanece activa mientras se ejecuta la subrutina de atención al periférico conectado al controlador. Luego de ser reconocida la interrupción se genera un pulso a '0', de un período de reloj de duración, en cada una de las salidas M1_out_n y IORQ_out_n. Estas pueden ser conectadas un periférico diseñado para modo 1 para darle el reconocimiento de interrupción.

4. Estructura Interna

El controlador está compuesto por cuatro bloques: CONTROL, detector_RETI, PUERTOS y control_INTA_modo1. El diagrama de bloques es mostrado en la figura .

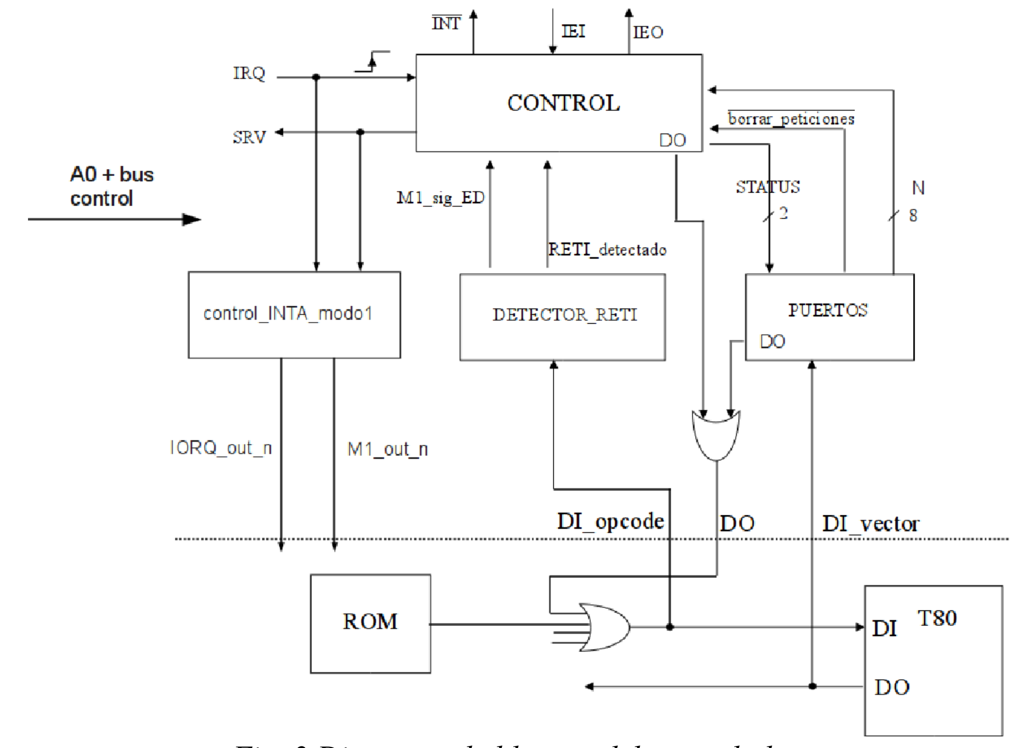


Fig. 2 Diagrama de bloques del controlador.

4.1 Bloque CONTROL

Recibe las solicitudes del periférico y maneja la salida SRV. Maneja las señales de protocolo IEI, IEO. Maneja la salida INT. Escribe el vector de interrupciones durante un ciclo INTA en su salida DO. Tiene una salida STATUS que da información sobre su estado .

4.2 Bloque detector_RETI

Observa el bus de entrada al T80 en su entrada DI opcode. Cuando se lee el opcode de la instrucción RETI (ED 4D) genera inmediatamente un pulso a '1' durante un período de reloj en su salida RETI_detectado.

4.3 Bloque PUERTOS

Implementa los puertos de entrada/salida. Mantiene en su salida N el vector de interrupciones para el bloque CONTROL. Cuando se escribe en el puerto de borrado de solicitudes pendientes genera un pulso a '0' en su salida borrar_pendientes hacia el bloque CONTROL. Lee de la salida del T80 en su entrada DI_vector y escribe en su salida DO.

4.4 Bloque control_INTA_modo1

Maneja las señales M1_out_n e IORQ_out_n según fue explicado en el punto **3**.

5. Operaciones de entrada/salida

La escritura y lectura del vector de interrupciones, lectura del estado del controlador y el borrado de peticiones pendientes son realizados mediante ciclos de entrada/salida. La decodificación se realiza con las entradas CE_n y A0 según indica la Tabla 1.

CE_n	A0	Escritura	Lectura
0	**0**	**N**	**N**
0	**1**	**borrar peticiones**	**ESTADO**

Tabla 1. Esquema de decodificación.

El borrado de peticiones se realiza independientemente del byte escrito en la dirección correspondiente. El byte N es el vector de interrupciones. Los 6 bits más significativos del byte ESTADO son cero, mientras que los 2 menos significativos codifican el estado del controlador de acuerdo a la Tabla 2:

Estado del controlador	Bits menos significativos de ESTADO
Sin solicitudes pendientes	"01"
Con solicitud pendiente	"10"
Subrutina en servicio	"11"

Tabla 2. Codificación del estado.

6. Descripción de pines

- **CK.** *Reloj del sistema.* El diseño fue validado para un ciclo de trabajo de 50%.
- **RESET_n.** *Reset del sistema* (entrada, activa por nivel bajo). Entrada de control.
- **M1_n.** *Machine Cycle One* (entrada, activa por nivel bajo). Entrada de control.
- **IORQ_n.** *Input/Output Request* (entrada, activa por nivel bajo). Entrada de control.
- **MREQ_n.** *Memory Request* (entrada, activa por nivel bajo). Entrada de control.
- **RD_n.** *Read* (entrada, activa por nivel bajo). Entrada de control.
- **WR_n.** *Write* (entrada, activa por nivel bajo). Entrada de control.
- **INT_n.** *Interrupt Request* (salida, activa por nivel bajo). Salida hacia la entrada $\overline{\text{INT}}$ del T80.
- **A0.** *Address 0* (entrada, activa por nivel alto). Bit menos significativo del bus de direcciones del T80.
- **DI_vector[7..0].** *Bus de entrada para escritura del vector de interrupciones* (entrada). De este bus se lee el vector de interrupciones cuando es programado.
- **DI_opcode[7..0].** *Bus de entrada para lectura de opcode* (entrada). El opcode de las instrucciones que ingresan al microprocesador es leído de este bus.
- **DO[7..0].** *Bus de salida* (salida). En este bus se colocan los datos de salida del controlador. Se utiliza en ciclos de entrada/salida. Durante ciclos de reconocimiento de interrupción el vector de interrupciones es colocado en este bus.
- **CE_n.** *Chip Enable* (entrada, activa por nivel bajo). Cuando está activa el controlador reconoce los ciclos de entrada/salida.
- **IEI.** *Interrupt Enable Input* (entrada, activa por nivel alto). Esta entrada activa indica que el dispositivo está habilitado para solicitar interrupciones.
- **IEO.** *Interrupt Enable Output* (salida, activa por nivel alto). Es desactivada cuando el controlador tiene una solicitud pendiente o en servicio.
- **IRQ.** *Interrupt Request* (entrada). Un flanco creciente en esta entrada indica una solicitud de interrupción del periférico conectado al controlador.
- **SRV.** *Servicio* (salida, activa por nivel alto). Activa mientras se está ejecutando la subrutina de atención del periférico conectado al controlador.

- **M1_out_n.** *Machine Cycle One Output* (salida, activa por nivel bajo). Salida para conectar a la entrada M1_n de un dispositivo modo 1. Cuando se solicita una interrupción se activa junto con la salida IORQ_out_n sólo cuando el ciclo de reconocimiento del T80 corresponda al dispositivo.

- **IORQ_out_n.** *Input/Output Request Output* (salida, activa por nivel bajo). Salida para conectar a la entrada IORQ_n de un dispositivo modo 1. Cuando se solicita una interrupción se activa junto con la salida M1_out_n sólo cuando el ciclo de reconocimiento del T80 corresponda al dispositivo.

7. Integración a un sistema con microprocesador T80

En esta sección son descriptas las conexiones que deben realizarse para integrar el controlador a un sistema basado en un microprocesador T80. Debe conectarse un controlador a cada periférico que pueda interrumpir.

- **Conexiones entre controladores:** La salida IEO de un controlador debe ser conectada a la entrada IEI del controlador con prioridad inmediatamente inferior. La entrada IEI del controlador de más prioridad debe ser conectada a '1'. No debe conectarse la salida IEO del controlador de menor prioridad.
- **Conexiones con el T80:** M1_n, IORQ_n, MREQ_n, RD_n, WR_n y A0 son conectados a los pines del T80 del mismo nombre.

  Los buses DI_vector, DI_opcode y DO deben conectarse como se indica en la Figura 2. No es necesario crear una habilitación para DO, puesto que el controlador mantiene este bus inactivo salvo durante los ciclos dirigidos a él.
- **Conexiones con un periférico:** El periférico debe poder generar un flanco creciente en la entrada IRQ para solicitar una interrupción. Si es un periférico modo 1 las salidas M1_out_n e IORQ_out_n deben conectarse a las entradas M1 e IORQ del periférico.
- CK y RESET_n se conectan a las señales de reloj y reset del sistema.
- La entrada CE_n debe ser conectada a la salida del circuito de decodificación elegido. El uso de la salida SRV es opcional.

8. Diagramas de tiempo

8.1 Solicitud de interrupción

En la figura 3 se muestra el diagrama de tiempos correspondiente a la solicitud de una interrupción por el periférico conectado al controlador. Luego del flanco creciente en IRQ la salida IEO es desactivada y se activa la salida INT_n.

8.2 Reconocimiento de interrupción

En la figura 4 se muestra el diagrama de tiempos correspondiente al comportamiento del controlador frente a un ciclo de reconocimiento de interrupción. Al activarse simultáneamente M1_n e IORQ_n con IEI = '1' es colocado el vector de interrupciones en DO. La salida SRV es activada. Obsérvese el pulso a '0' en las salidas M1_out_n e IORQ_out_n.

8.3 Retorno desde una subrutina

En la figura 5 se muestra el diagrama de tiempos cuando es leída la instrucción RETI en DI_opcode e IEI = '1'. Luego de esta instrucción se desactiva SRV y se activa IEO.

8.4 Lectura del estado

En la figura 6 se muestra el diagrama de tiempos de un ciclo de lectura del estado del controlador. En este ciclo de lectura es A0 = '1'.

8.5 Lectura/escritura del vector de interrupciones

En las figuras 7 y 8 se muestran los diagramas de tiempos los ciclos de escritura y lectura del vector de interrupciones. En ambos casos es A0 = '0'.

8.6 Borrado de solicitudes pendientes

En la figura 9 se muestra el borrado de una solicitud pendiente mientras está en servicio una solicitud anterior.

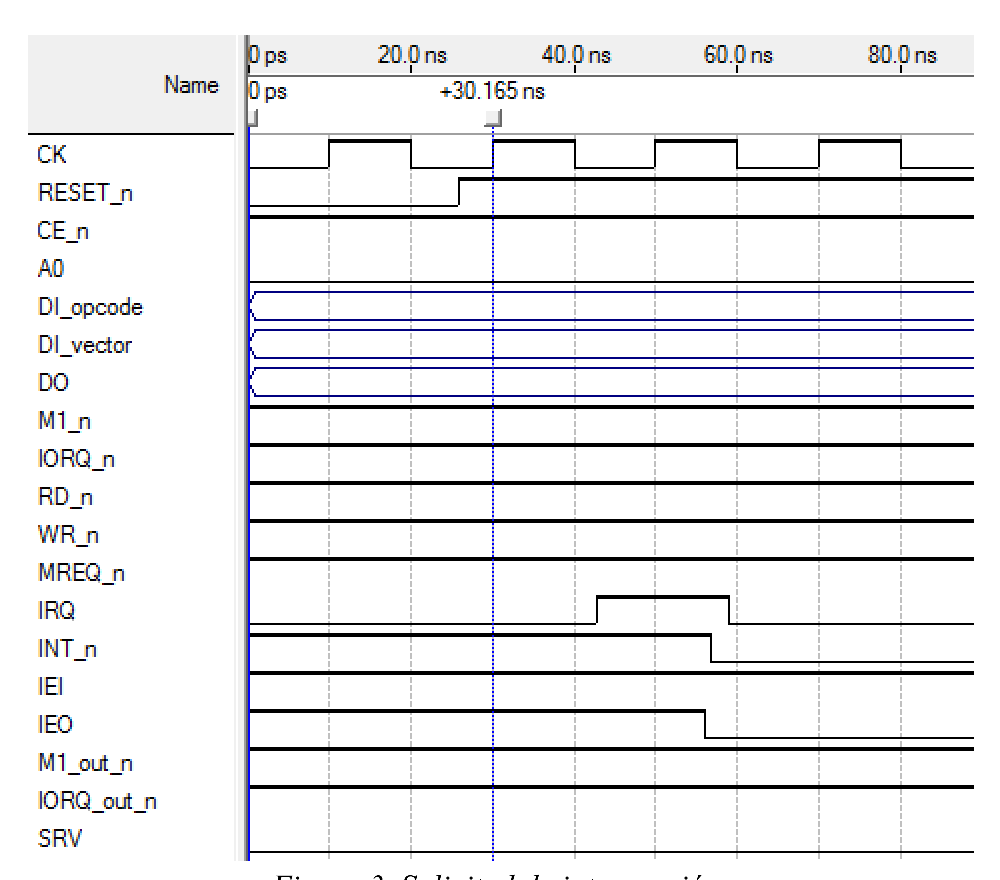


Figura 3. Solicitud de interrupción.

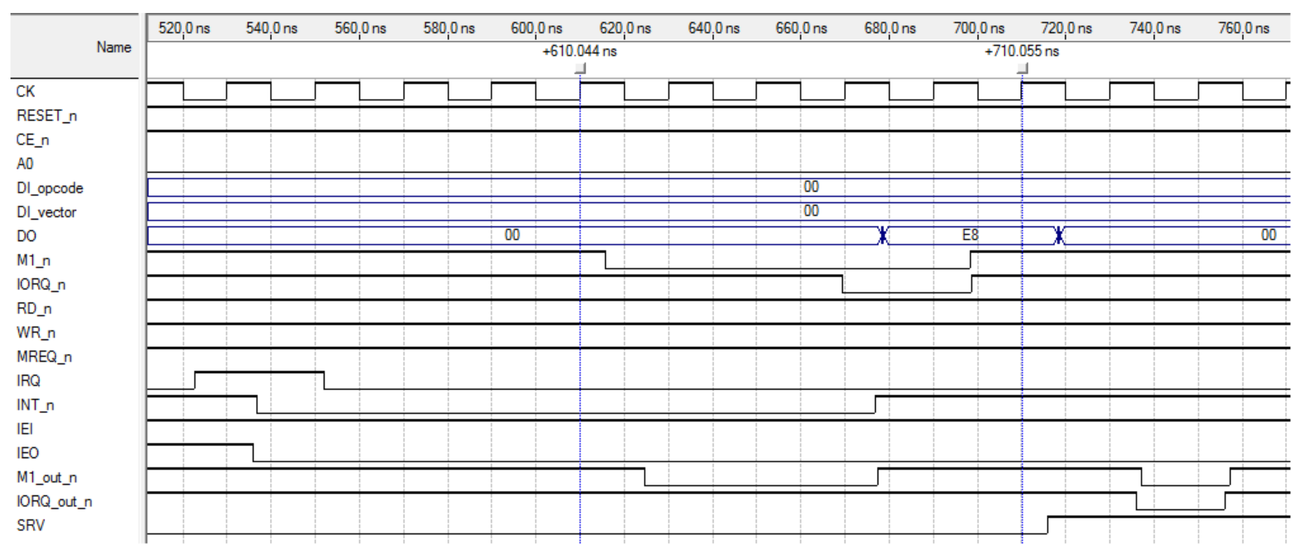


Figura 4. Ciclo de reconocimiento de interrupción.

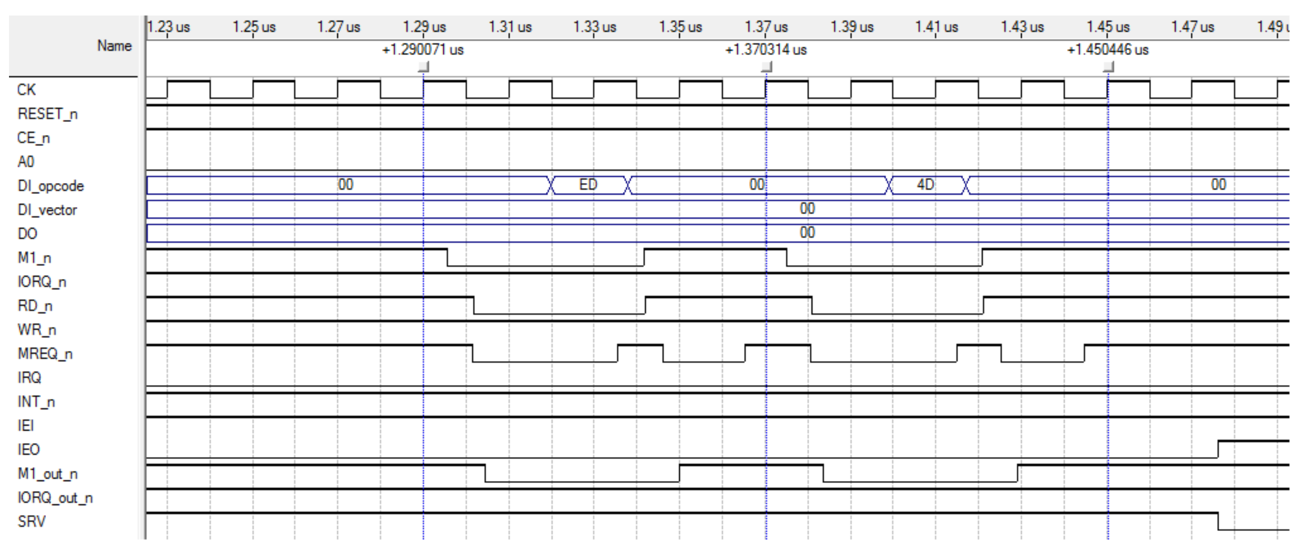


Figura 5. Retorno desde una subrutina.

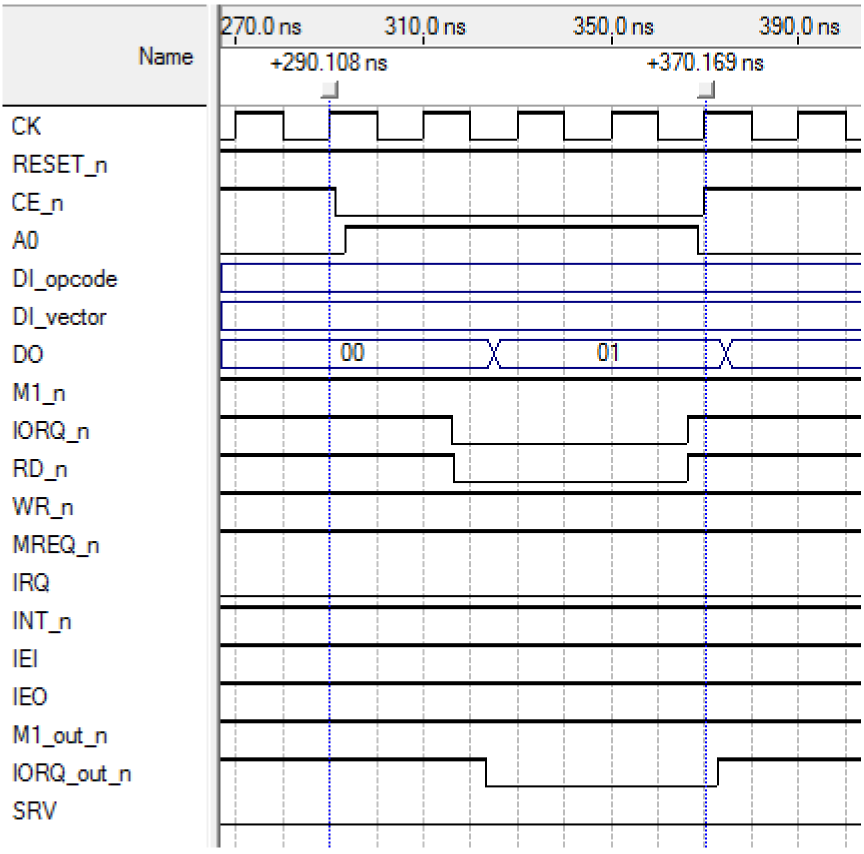


Figura 6. Lectura de estado.

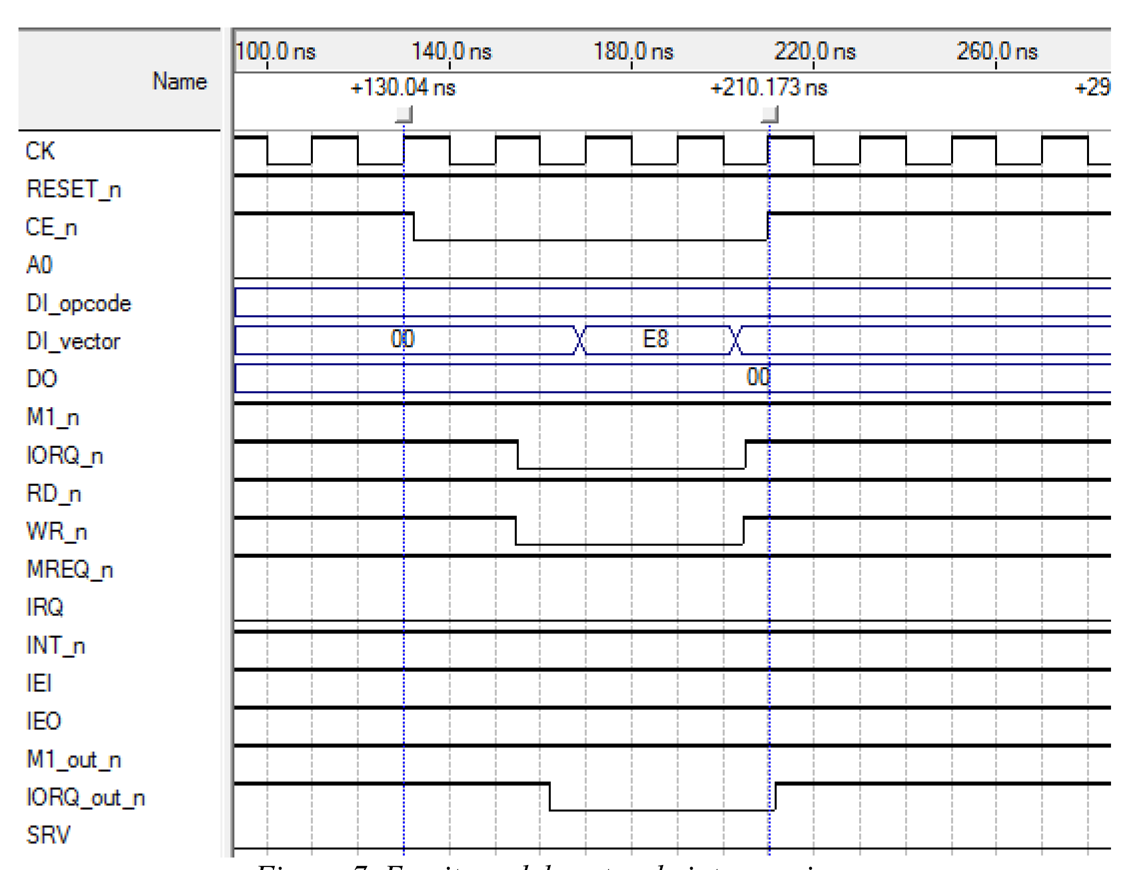


Figura 7. Escritura del vector de interrupciones.

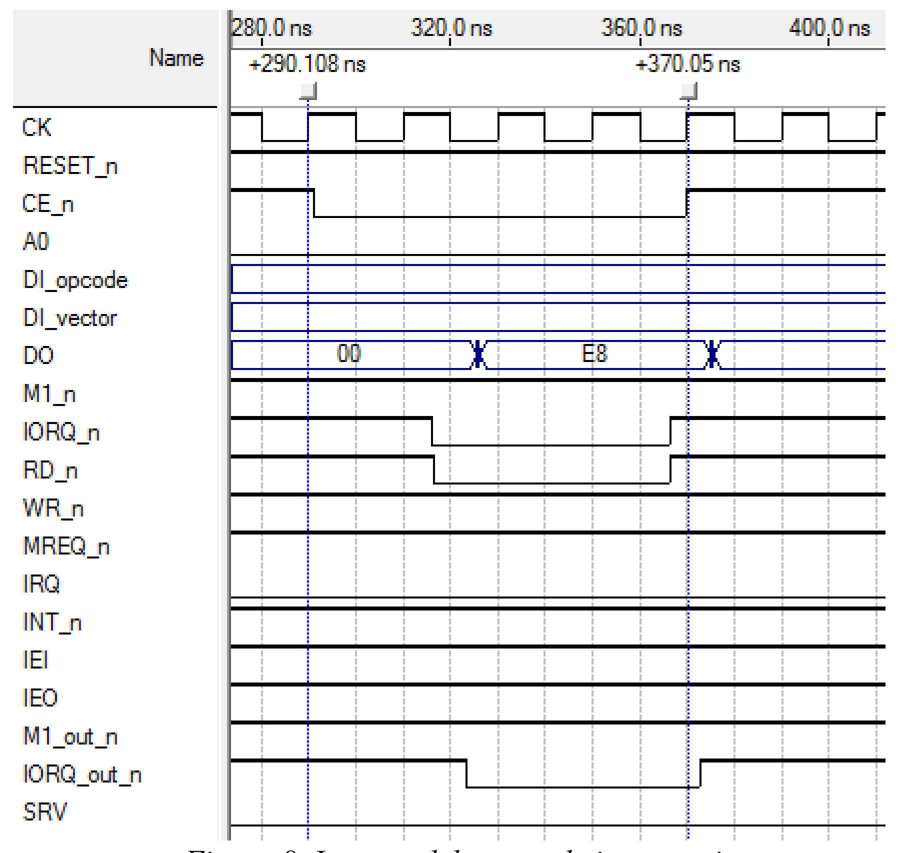


Figura 8. Lectura del vector de interrupciones.

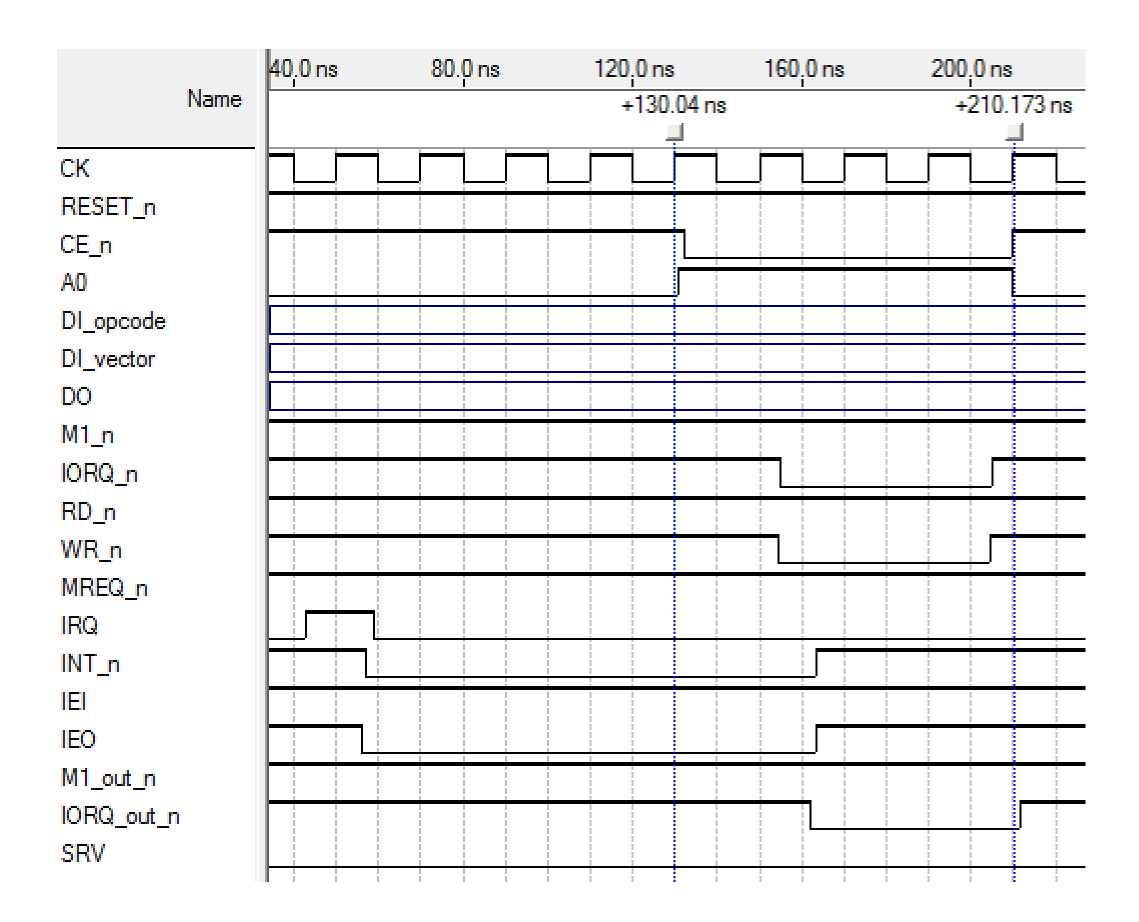


Figura 9. Borrado de una solicitud pendiente.